CLAIMS

What is claimed is:

1. A memory comprising:

a memory array to store data;

a first pointer coupled to the memory array to address memory locations therein;

a pointer memory coupled to the first pointer, the pointer memory to save one or more prior first pointer values of the first pointer; and

control logic coupled to the pointer memory, the control logic to restore one of the one or more prior first pointer values to the first pointer in response to branch information.

2. The memory of claim 1, wherein

the first pointer is a pop pointer to read data out from the memory array,

and

the memory further comprises,

a push pointer to write data into the memory array.

3. The memory of claim 1, wherein

the branch information includes a branch flag to indicate a condition that requires the restoration of the one prior first pointer value to the first pointer.

4. The memory of claim 1, wherein

the branch information is to be received from a processor capable of speculatively issuing requests to read data from the memory array.

5. The memory of claim 1, wherein

the pointer memory saves a plurality of prior first pointer values as a history of first pointer values.

6. The memory of claim 2, further comprising:

status logic coupled to the pop pointer and the push pointer to monitor a pop pointer value of the pop pointer and a push pointer value of the push pointer to provide an indication of an amount of data stored in the memory array.

7. The memory of claim 6, wherein

the status logic to generate a high status flag in response to an amount of data stored in the memory array being greater than or equal to a high threshold level, and less than or equal to a maximum utilization level.

8. The memory of claim 7, wherein

the status logic to further generate a low status flag in response to an amount of data stored in the memory array being less than or equal to a low threshold level and greater than or equal to an empty threshold level.

9. The memory of claim 7, wherein

the high threshold level is responsive to the lesser of a maximum branch resolution latency and a low threshold level.

10. The memory of claim 9, wherein

the maximum branch resolution latency is a depth of an instruction pipeline in a processor, the processor to couple to the first-in first-out memory.

11. The memory of claim 1, wherein

the branch information includes a branch resolution latency, the branch resolution latency is the number of instruction cycles to resolve a conditional branch instruction in a processor, the processor to couple to the memory.

12. The memory of claim 9, wherein

the memory array is capable of being directly addressed by a processor to randomly access storage locations therein.

13. The memory of claim 2, wherein
the memory is a first-in first-out memory.

14. A method for a first-in first-out (FIFO) memory, the method comprising:
storing one or more prior pop pointer values of a pop pointer;
processing one or more pop requests to read data from the FIFO memory;
receiving information to indicate at least one of the one or more pop requests was speculative and a state of the pop pointer of the FIFO memory should be restored; and
restoring one of the one or more prior pop pointer values to the pop pointer in response to the information.

15. The method of claim 14, wherein
the one or more prior pop pointer values of the pop pointer are stored into a pointer memory.

16. The method of claim 15, wherein
the restoring of the one prior pop pointer value to the pop pointer includes
reading the one prior pop pointer value from the pop pointer memory, and

loading the one prior pop pointer value into the pop pointer.

17. The method of claim 14, further comprising:

prior to the processing of the one or more pop requests,

storing data into a memory array of the FIFO memory, and

incrementing a push pointer.

18. The method of claim 17, further comprising:

reading a pop pointer value of the pop pointer and a push pointer value of the push pointer, and

determining a status of the memory array in response to the pop pointer value and the push pointer value.

19. The method of claim 18, wherein

the determining of the status of the memory array is further in response to a high threshold level and a low threshold level.

20. The method of claim 19, wherein

the high threshold level is responsive to the lesser of a maximum branch resolution latency and the low threshold level.

21. The method of claim 20, wherein

the maximum branch resolution latency is a depth of an instruction pipeline in a processor, the processor to couple to the first-in first-out memory.

22. The method of claim 20, wherein

the branch information includes a branch resolution latency, the branch resolution latency is the number of instruction cycles to resolve a conditional branch instruction in a processor, the processor to couple to the first-in first-out memory.

23. The method of claim 17, further comprising:

bypassing the pop pointer and the push pointer, and directly addressing the memory array of the FIFO memory to read or write data thereto.

24. The method of claim 17, further comprising:

loading the pop pointer with an address of the memory array to randomly read data there-from.

25. The method of claim 17, further comprising:

loading the push pointer with an address of the memory array to randomly write data thereto.

26. A data signal flow coupled into a first-in first out (FIFO) memory, the data signal flow comprising:

branch resolution latency to determine which one of one or more prior pointer values to restore into a state of a pop pointer; and

a branch indicator to indicate that a conditional branch instruction was resolved to take the branch.

27. The data signal flow of claim 26, wherein

the branch resolution latency is a number indicating how many instruction cycles were executed to resolve a conditional branch instruction in a processor, the processor to couple to the first-in first-out memory.

28. The data signal flow of claim 26, wherein

the branch resolution latency is a number indicating a depth of an instruction pipeline in a processor, the processor to couple to the first-in first-out memory

29. The data signal flow of claim 26, further comprising:

a delayed branch indicator to indicate a delay of a branch instruction.

30. The data signal flow of claim 29, wherein

the delayed branch indicator indicates the delay of the branch instruction is zero.

31. The data signal flow of claim 26, further comprising:

one or more pop requests to read data from a memory array of the FIFO memory.

32. A processing unit including:

a plurality of processors, each of the processors including an instruction pipeline to speculatively execute instructions before a conditional branch is resolved;

a first plurality of branch-aware first-in first-out (FIFO) memories to pass data from one processor to the next in a first direction, each branch-aware FIFO memory of the first plurality of branch-aware FIFO memories interleaved between a pair of processors of the plurality of processors;

a first input branch-aware FIFO memory coupled to a first processor of the plurality of processors to receive input data in the processing unit; and

a first output FIFO memory coupled to a last processor of the plurality of processors to drive output data from the processing unit.

33. The processing unit of claim 32, further comprising:

a second plurality of branch-aware first-in first-out (FIFO) memories to pass data from one processor to the next in a second direction, each branch-aware FIFO memory of the second plurality of branch-aware FIFO

memories interleaved between a pair of processors of the plurality of processors;

a second input branch-aware FIFO memory coupled to a last processor of the plurality of processors to receive input data in the processing unit; and

a second output FIFO memory coupled to the first processor of the plurality of processors to drive output data from the processing unit.

34. The processing unit of claim 32, wherein each branch-aware FIFO memory includes,

a memory array to store data;

a push pointer coupled to the memory array to address memory locations therein to write data;

a pop pointer coupled to the memory array to address memory locations therein to read data;

a pointer memory coupled to the pop pointer, the pointer memory to save one or more prior pop pointer values of the pop pointer; and

control logic coupled to the pointer memory, the control logic to restore one of the one or more prior pop pointer values to the pop pointer in response to branch information received from a processor.

35. The processing unit of claim 33, wherein each branch-aware FIFO memory includes,

a memory array to store data;

a push pointer coupled to the memory array to address memory locations therein to write data;

a pop pointer coupled to the memory array to address memory locations therein to read data;

a pointer memory coupled to the pop pointer, the pointer memory to save one or more prior pop pointer values of the pop pointer; and

control logic coupled to the pointer memory, the control logic to restore one of the one or more prior pop pointer values to the pop pointer in response to branch information received from a processor.

36. A computer system including:

an input/output device;

dynamic random access memory; and

a multi-processor coupled to the dynamic random access memory and the input/output device, the multi-processor including,

a plurality of processors, each of the processors including an instruction pipeline to speculatively execute instructions before a conditional branch is resolved;

a first plurality of branch-aware first-in first-out (FIFO) memories to pass data from one processor to the next in a first direction, each

branch-aware FIFO memory of the first plurality of branch-aware FIFO memories interleaved between a pair of processors of the plurality of processors;

a first input branch-aware FIFO memory coupled to a first processor of the plurality of processors to receive input data in the processing unit;

a first output FIFO memory coupled to a last processor of the plurality of processors to drive output data from the processing unit; and

wherein each branch-aware FIFO memory includes,

a memory array to store data,

a push pointer coupled to the memory array to address memory locations therein to write data,

a pop pointer coupled to the memory array to address memory locations therein to read data,

a pointer memory coupled to the pop pointer, the pointer memory to save one or more prior pop pointer values of the pop pointer, and

control logic coupled to the pointer memory, the control logic to restore one of the one or more prior pop pointer values to the pop pointer in response to branch information received from a processor.

37. The computer system of claim 36, wherein the multi-processor further includes,

a second plurality of branch-aware first-in first-out (FIFO) memories to pass data from one processor to the next in a second direction, each branch-aware FIFO memory of the second plurality of branch-aware FIFO memories interleaved between a pair of processors of the plurality of processors;

a second input branch-aware FIFO memory coupled to a last processor of the plurality of processors to receive input data in the processing unit; and

a second output FIFO memory coupled to the first processor of the plurality of processors to drive output data from the processing unit.

38. A processor comprising:

an instruction pipeline to speculatively execute instructions before a conditional branch is resolved; and

a first branch-aware first-in first-out (FIFO) memory to pass data from the processor to another processor, the first branch-aware FIFO memory to receive branch information responsive to the conditional branch, the first branch-aware FIFO memory including

a memory array to store data,

a push pointer coupled to the memory array to address memory locations therein to write data,

a pop pointer coupled to the memory array to address memory locations therein to read data,

a pointer memory coupled to the pop pointer, the pointer memory to save one or more prior pop pointer values of the pop pointer, and

control logic coupled to the pointer memory, the control logic to restore one of the one or more prior pop pointer values to the pop pointer in response to the branch information.

39. The processor of claim 38, wherein

the branch information includes a branch flag to indicate a condition requiring restoration of the one prior pop pointer value to the pop pointer.

40. The processor of claim 38, wherein

the pointer memory saves a plurality of prior pop pointer values as a history of pop pointer values.

41. The processor of claim 38, further comprising:

status logic coupled to the pop pointer and the push pointer to monitor a pop pointer value of the pop pointer and a push pointer value of the push pointer to provide an indication of an amount of data stored in the memory array.

42. The processor of claim 41, wherein

the status logic to generate a high status flag in response to an amount of data stored in the memory array being greater than or equal to a high threshold level, and less than or equal to a maximum utilization level.

43. The processor of claim 41, wherein

the status logic to further generate a low status flag in response to an amount of data stored in the memory array being less than or equal to a low threshold level and greater than or equal to an empty threshold level.

44. The processor of claim 42, wherein

the high threshold level is responsive to the lesser of a maximum branch resolution latency and a low threshold level.

45. The processor of claim 44, wherein

the maximum branch resolution latency is a depth of an instruction pipeline in a processor, the processor to couple to the first-in first-out memory.

46. The processor of claim 38, wherein

the branch information includes a branch resolution latency, the branch resolution latency is the number of instruction cycles to resolve a conditional branch

instruction in a processor, the processor to couple to the memory.

47. The processor of claim 38, wherein the memory array is capable of being directly addressed to randomly access storage locations therein.

48. The processor of claim 38, further comprising: a second branch-aware first-in first-out (FIFO) memory to pass data from another processor to the processor.

49. The processor of claim 48, wherein the second branch-aware FIFO memory to receive branch information responsive to a second conditional branch, and the second branch-aware FIFO memory includes a second memory array to store data, a second push pointer coupled to the second memory array to address memory locations therein to write data, a second pop pointer coupled to the second memory array to address memory locations therein to read data, a second pointer memory coupled to the second pop pointer, the second pointer memory to save one

or more prior second pop pointer values of the second pop pointer, and

a second control logic coupled to the second pointer memory, the second control logic to restore one of the one or more prior second pop pointer values to the second pop pointer in response to the second branch information.